

RANCANG BANGUN EMAIL CLIENT PADA PERANGKAT MOBILE

Tri Aditya Sasongko, Idris Winarno, S.ST, M.Kom, Kholid Fathoni, S.Kom.

Jurusan Teknik Informatika, Politeknik Elektronika Negeri Surabaya, Institut Teknologi Sepuluh Nopember,

Email: adityasasongko@live.com

Email merupakan salah satu fasilitas yang sering digunakan orang karena memberikan cara yang mudah dan cepat dalam mengirimkan sebuah informasi. Dengan menggunakan email client, orang dapat mengakses email dengan mudah dan cepat. Kebutuhan masyarakat dalam menggunakan email ini didukung dengan perkembangan teknologi informasi yang semakin pesat telah menempatkan mobile devices khususnya mobile phone sebagai salah satu alat komunikasi yang efisien untuk mengakses, menyimpan dan memproses informasi. Sebuah email yang dapat diakses dimanapun dengan menggunakan media perangkat mobile merupakan salah satu opsi yang mempunyai manfaat yang cukup besar, karena manfaat akan email itu sendiri yang seringkali diakses oleh pengguna untuk saling berkomunikasi.

Pada Tugas Akhir ini membuat suatu email client yang dapat dijalankan pada perangkat mobile dengan dukungan Java dan koneksi GPRS. Dengan aplikasi ini, Pengguna dapat mengakses dan memanipulasi pesan-pesan yang disimpan di POP3 dan IMAP server. Akses dan manipulasi yang dapat dilakukan antara lain melihat daftar pesan, membaca pesan, menghapus pesan, dan menyimpan pesan. Selain itu, Pengguna juga dapat mengirim pesan melalui SMTP server.

Kata Kunci : *Email Client, Mobile Phones, Java ME*

1. PENDAHULUAN

Email sudah digunakan orang sejak awal terbentuknya internet dan merupakan salah satu fasilitas yang ada pada saat itu. Sesuai dengan perkembangan internet, penggunaan email ini juga semakin membesar. Salah satu alasan kenapa email dipakai orang karena memberikan cara yang mudah dan cepat dalam mengirimkan sebuah informasi.

Email bergerak dari suatu komunikasi yang opsional menjadi esensial. Hal ini terbukti jika kita lihat etos kerja para pekerja di kota besar. Setiap hari, tiap kali membuka computer, akun email merekalah yang dicek. Entah sekedar mengikuti milis atau menjadi alat komunikasi bisnis, email sudah tidak terelakkan lagi menjadi alat komunikasi yang banyak dipakai.

Pada Tugas Akhir ini dibangun sebuah aplikasi email client menggunakan teknologi Java ME. Dengan aplikasi ini, pengguna dapat mengakses dan memanipulasi pesan-pesan yang disimpan di mail server. Akses dan manipulasi yang dapat dilakukan antara lain melihat daftar pesan, membaca pesan,

menghapus pesan dan menyimpan pesan. Selain itu, pengguna juga dapat mengirim pesan melalui SMTP server.

2. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dan manfaat dari proyek akhir ini adalah membuat suatu email client yang dapat dijalankan pada perangkat mobile dengan dukungan Java dan koneksi GPRS. Dengan aplikasi ini, Pengguna dapat mengakses dan memanipulasi pesan-pesan yang disimpan di POP3 dan IMAP server. Akses dan manipulasi yang dapat dilakukan antara lain melihat daftar pesan, membaca pesan, menghapus pesan, dan menyimpan pesan. Selain itu, Pengguna juga dapat mengirim pesan melalui SMTP server

3. TINJAUAN PUSTAKA

3.1 Email

Electronic-Mail (E-Mail) merupakan aplikasi TCP/IP yang paling banyak digunakan. Email adalah pesan yang terdiri atas kumpulan string ASCII dalam format RFC 822 (dikembangkan thn 1982).

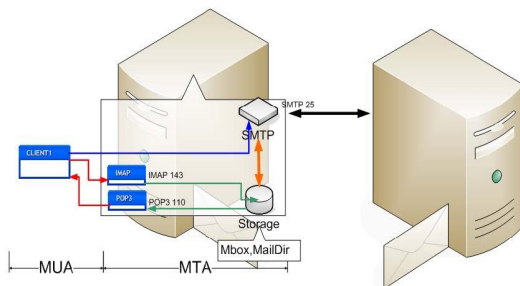
Sistem email yang beroperasi di atas jaringan berbasis pada model *store and forward*. Sistem ini mengaplikasikan sebuah sistem server email yang menerima, meneruskan, dan mengirimkan, serta menyimpan pesan-pesan user, dimana user hanya perlu untuk mengkoneksikan komputer mereka ke dalam jaringan.

Email dapat dianalogikan dengan kotak surat yang ada di kantor pos sedangkan server email dapat diibaratkan sebagai kantor pos. Dengan analogi ini sebuah mail server dapat memiliki banyak account email yang ada didalamnya.

Beberapa pengamat mempertanyakan apakah teknologi *Push* mirip dengan *e-mail* dan bukanlah sesuatu yang baru. Metode *Listservs* telah ada untuk pendistribusian pesan *e-mail* secara otomatis tanpa harus user melakukan request. Pada dasarnya prinsip dari *Listserv* mirip dengan teknologi *push*, teknologi *Push* mengijinkan distribusi informasi dan gambar yang lebih kompleks dari apa yang dapat ditangani oleh

e-mail dan dilakukan real-time dimana sistem *store and forwards* dari *e-mail* tidak di-desain untuk itu.

Cara kerja email dapat dilihat pada Gambar 3.1. E-mail yang dikirim belum tentu akan diteruskan ke komputer penerima (end user), tapi disimpan/dikumpulkan dahulu dalam sebuah komputer server (host) yang akan online secara terus menerus (continue) dengan media penyimpanan (storage) yang relatif lebih besar dibanding komputer biasa. Hal ini bisa diibaratkan dengan sebuah kantor pos, jika seseorang mempunyai alamat (mailbox), maka dia dapat memeriksa secara berkala jika dia mendapatkan surat. Komputer yang melayani penerimaan email secara terus-menerus tersebut biasa disebut dengan mailserver atau mailhost.



Gambar 3. 1 Cara kerja email

3.2 SMTP

Simple Mail Transfer Protocol (SMTP) merupakan salah satu protokol yang umum digunakan untuk pengiriman surat elektronik di Internet. Protokol ini dipergunakan untuk mengirimkan data dari komputer pengirim surat elektronik ke server surat elektronik penerima.

Protokol ini timbul karena desain sistem surat elektronik yang mengharuskan adanya server surat elektronik yang menampung sementara sampai surat elektronik diambil oleh penerima yang berhak.

Contoh transaksi dari SMTP adalah sebagai berikut :

```
S = Server
C = Klien

C : (inisialisasi hubungan)
S : 220 Wikipedia ESMTP server ready
C : mail from: user@wikipedia.org
S : 250 mail ok
C : rcpt to: wikipedia@wikipedia.org
S : 250 rcpt ok
C : data
S : 354 end data with CRLF.CRLF
C : -- email data --
C : CRLF.CRLF
S : 250 data ok
C : QUIT
```

S : 221 bye

3.3 POP3

Post Office Protocol version 3 (POP3) adalah protokol yang digunakan untuk mengambil surat elektronik (email) dari server email. Protokol ini erat hubungannya dengan protokol SMTP dimana protokol SMTP berguna untuk mengirim surat elektronik dari komputer pengirim ke server.

Protokol POP3 dibuat karena desain dari sistem surat elektronik yang mengharuskan adanya server surat elektronik yang menampung surat elektronik untuk sementara sampai surat elektronik tersebut diambil oleh penerima yang berhak. Kehadiran server surat elektronik ini disebabkan kenyataan hanya sebagian kecil dari komputer penerima surat elektronik yang terus-menerus melakukan koneksi ke jaringan internet. Protokol ini dispesifikasikan pada RFC 1939.

Contoh transaksi dari POP3 adalah sebagai berikut :

```
S = Server
C = Client

C : (inisialisasi koneksi)
S : +OK
C : USER wikipedia
S : +OK
C : PASS indonesia
S : +OK wikipedia have 2 message (320
octets)
C : STAT
S : +OK 2 320

C : LIST
S : 1 150
S : 2 170
S : .

C : RETR 1
S : +OK 150
S : --email data--
S : .
C : DELE 1
S : +OK
C : QUIT
S : +OK Bye
```

3.3 IMAp

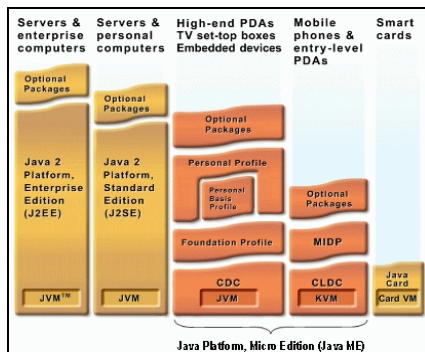
IMAP (Internet Message Access Protocol) adalah protokol standar untuk mengakses/mengambil e-mail dari server. IMAP memungkinkan pengguna memilih pesan e-mail yang akan ia ambil, membuat

folder di server, mencari pesan e-mail tertentu, bahkan menghapus pesan e-mail yang ada.

Kemampuan ini jauh lebih baik daripada POP (Post Office Protocol) yang hanya memperbolehkan kita mengambil/download semua pesan yang ada tanpa kecuali.

3.4 Java ME

Java ME adalah lingkungan pengembangan yang didesain untuk menggunakan aplikasi java pada peralatan elektronik kecil, seperti telepon seluler, PDA, dan sejenisnya. Java ME dibuat untuk mengatasi keterbatasan yang berhubungan dengan pembuatan aplikasi pada peralatan elektronik kecil. Karena itu teknologi Java ME ini disesuaikan dengan keterbatasan memori, tampilan dan tenaga. Hal tersebut dapat terlihat pada gambar 3.2 yang membandingkan teknologi Java ME dengan teknologi Java lainnya.



Gambar 3. 2 Perbandingan Java ME dengan teknologi Java lainnya.

Java ME berbasis pada 3 elemen yaitu :

- Konfigurasi, yang menyediakan library paling dasar dan kemampuan *virtual memory* untuk berbagai jenis peralatan elektronik. Terdapat dua buah konfigurasi pada Java ME, yaitu CLDC (*Connected Limited Device Configuration*) untuk peralatan yang kecil, dan CDC (*Connected Device Configuration*) untuk peralatan yang lebih besar.
- Profile, yang merupakan kumpulan API spesifik yang mendukung sebuah peralatan elektronik.
- Paket-paket tambahan yang berisi kumpulan API berbasis teknologi tertentu.

3.5 Record Management Store

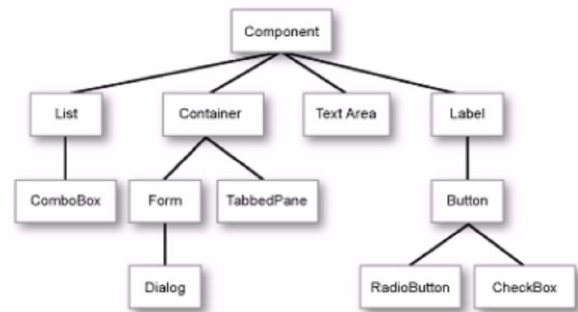
MIDP menyediakan sebuah API dimana program dapat menyimpan data-data aplikasi secara lokal didalam device tersebut. MIDP Record

Management System adalah sebuah fasilitas yang dimiliki oleh MIDlets untuk menyimpan data-data aplikasi pada saat MIDlet invocations. Data akan disimpan dalam non-volatile memory didalam device. Hal ini berarti, data-data program yang telah disimpan tidak akan hilang walaupun program di restart maupun device dimatikan.

3.6 LWUIT

Lightweight User Interface Toolkit (LWUIT) adalah lightweight widget library dari SUN Microsystem yang terinspirasi oleh swing namun di desain untuk device dengan kapasitas terbatas seperti mobile phone dan PDA. LWUIT mendukung kemampuan pluggable theme, hirarki komponen dan kontainer serta abstraksi dibawah GUI toolkit.

Interface internal dan kelas abstrak menyediakan abstraction dari interface dan API pada profile sehingga memudahkan portability dan migrasi untuk device dan profile di masa mendatang. LWUIT menghindari lowest common denominator yakni mengimplementasikan beberapa fitur pada platform low-end dan membawa hasil yang lebih baik pada platform high-end. Struktur komponen dari LWUIT ditunjukkan pada gambar 3.3



Gambar 3. 3 Struktur komponen LWUIT

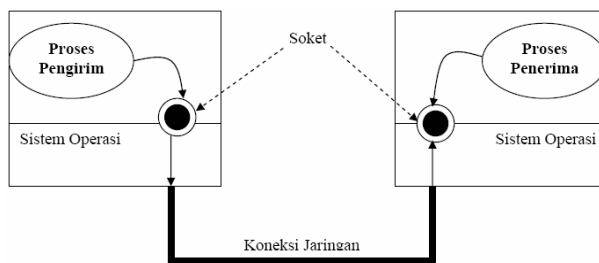
LWUIT merupakan sebuah library untuk widget UI dan tidak mengatur tentang bagaimana sistem pada device seperti *networking* atau *storage*. Hal hal yang berkaitan dengan sistem sudah diatur oleh Java ME *profile* sebab LWUIT pada dasarnya adalah sebuah UI *framework* yang menawarkan antar muka yang lebih baik, teratur dan mudah untuk diimplementasikan oleh developer.

Untuk membuat menjadi portable, LWUIT mengimplementasikan layer tipis diatas *native system* dari kanvas dan menyediakan sebuah widget abstraction. Abstraction tersebut digunakan dengan memanfaatkan kelas-kelas yang

ada di dalam sistem seperti *Graphic, Image* dan *Font*.

3.7 Socket

Socket merupakan fasilitas IPC (*Inter Proses Communication*) untuk aplikasi jaringan. Agar suatu socket dapat berkomunikasi dengan socket lainnya, maka socket butuh diberi suatu alamat unik sebagai identifikasi. Alamat socket terdiri atas alamat IP dan nomor port. Contoh alamat socket adalah 192.168.29.30:3000, dimana nomor 3000 adalah nomor portnya. Alamat IP dapat menggunakan alamat jaringan lokal (LAN) maupun alamat internet. Jadi socket dapat digunakan untuk IPC pada LAN maupun Internet.



Gambar 3.4 Socket

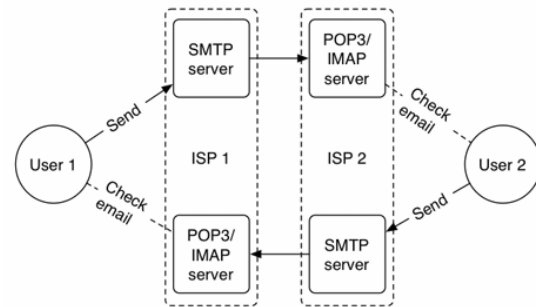
4. Perancangan Perangkat Lunak

4.1 Deskripsi Umum Perangkat Lunak

Perangkat lunak mobile email client ini dinamakan James (*Javamail Midlet Application*) adalah sebuah perangkat lunak yang memiliki arsitektur client-server yang bertindak sebagai client yang terletak pada perangkat mobile. Perangkat lunak ini akan berkomunikasi dengan email server pada jaringan internet melalui protokol SMTP, IMAP, dan POP3.

James berkomunikasi dengan mail server melalui koneksi data internet yang didukung oleh perangkat mobile (misal: GPRS).

James berfungsi sebagai penghubung antara pengguna dengan email server. Perangkat lunak ini akan meneruskan perintah dari pengguna yang kemudian dikirim ke email server tujuan. Kemudian perangkat lunak ini akan mendapat respon dari email server dan menampilkan respon ke layar sesuai dengan kebutuhan pengguna. James memiliki antarmuka bagi pengguna dilengkapi dengan fungsi-fungsi untuk mengakses email. Sehingga mempermudah user dalam menggunakan perangkat lunak ini.



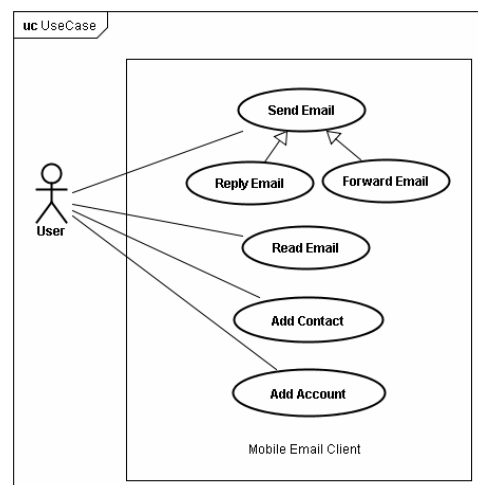
Gambar 4.1 Arsitektur Sistem

Skenario pada sistem bisnis dalam aplikasi ini adalah sebagai berikut :

- Pengguna dapat menerima dan membaca pesan yang diterima dari server.
- Pengguna dapat mengirimkan pesan.
- Pengguna dapat memfilter pesan yang dimilikinya.
- Pengguna dapat me-reply atau mem-forward pesan ke pada pengguna lain.
- Pengguna dapat menambahkan daftar alamat pengguna lain.
- Pengguna dapat menambahkan daftar account.

4.2 Use case diagram

Diagram use case pada gambar 4.2 menggambarkan bahwa aplikasi James ini memiliki 1 aktor yakni user. User dapat melakukan mengirim email dan membaca email yang masuk. User juga dapat manajemen contact yakni menambahkan atau menghapus contact serta user juga dapat manajemen setting yakni memasukkan account email atau menghapusnya

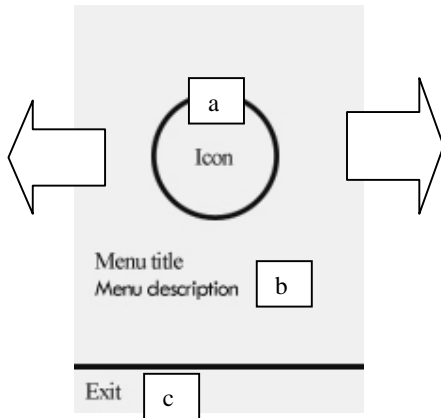


Gambar 4.2 Use Case Diagram

4.3 Antar Muka

▪ Antar Muka halaman utama.

Halaman ini adalah halaman utama dari aplikasi. Halaman ini terdiri dari 5 menu utama yaitu : *Mail, Account, Contact, dan Help*. Untuk berpindah dari satu menu ke menu lain, user dapat menekan tombol kiri dan kanan pada *mobile device* nya karena halaman menu utama dari aplikasi ini di desain menggunakan teknik *sliding*.



Gambar 4. 3 Rancangan Antar Muka Halaman Menu Utama

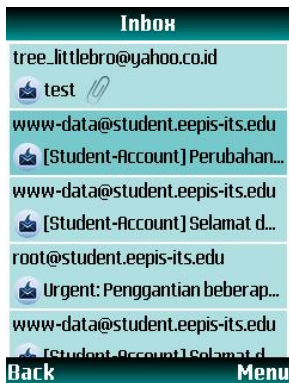
Penjelasan gambar 4.3 adalah :

- Icon dari menu
- Keterangan dari menu, terdiri dari *menu title* dan *menu description*.
- Exit Command*

5. UJI COBA

Untuk uji coba akan menggunakan mail server *sinau.eepis-its.edu* dan *student.eepis-its.edu*.

Skenario dari uji coba adalah memasukkan account dari email. Hasil uji coba untuk *retrieve* email dapat dilihat pada gambar 5.1.



Gambar 5.1 Inbox hasil *retrieve* email

6. KESIMPULAN

Dari pelaksanaan uji coba pada bab sebelumnya, didapatkan kesimpulan sebagai berikut :

1. Aplikasi Email Client ini dirancang sebagai aplikasi client untuk menerima, mengirim, ndan memfilter message dari mail server.
2. Dalam pengiriman dan penerimaan message digunakan protokol socket untuk berkomunikasi dengan mail server.
3. Untuk dapat mengunduh pesan dan mengirim pesan, pengguna harus membuat account dan mengaktifkannya pada menu Add Account.
4. Pengguna mobile email client dapat mengelompokkan(memfilter) pesan sesuai dengan kata maupun kalimat yang terkandung dalam email header kedalam sebuah folder.
5. Pengguna mobile mail client dapat mengirim, menerima, dan membaca pesan yang telah diunduh dari mail server.

7. DAFTAR PUSTAKA

- [1] <http://developers.sun.com/mobility/midp/articles/midp2network/>
- [2] <http://id.wikipedia.org/wiki/SMTP> (10 januari 2010)
- [3] <http://id.wikipedia.org/wiki/POP3> (10 januari 2010)
- [4] <http://id.wikipedia.org/wiki/IMAP4> (10 januari 2010)
- [6] <http://ebusiness.gbdirect.co.uk/howtos/mail-system.html> (10 januari 2010)
- [7] <http://www.dreamincode.net/forums/forum/109-mobile-development/>
- [8] <http://tech.groups.yahoo.com/group/jug-indonesia/>
- [9] <https://lwuit.dev.java.net/>
- [10] <http://java.sun.com/javame/index.jsp>
- [11] <http://tools.ietf.org/html/rfc2821> (RFC - SMTP)

- [12] <http://tools.ietf.org/html/rfc1939> (RFC - POP3)
- [13] <http://tools.ietf.org/html/rfc3501> (RFC - IMAP)
- [14] Sarkar, Biswajit, 2009, “LWUIT 1.1 for Java ME Developer”, Packt Publishing.

8. DAFTAR RIWAYAT HIDUP



Penulis dilahirkan di kota Surabaya, Jawa Timur, 21 tahun silam. Merupakan anak ketiga dari pasangan Bapak Eddy Malik dan Ibu Yurinarni.

Riwayat Pendidikan yang pernah ditempuh oleh penulis : SDN Wonokromo IV Surabaya, SLTP Negeri 32 Surabaya, SMU Negeri 4 Surabaya kemudian melanjutkan ke D4 Teknik Informatika Politeknik Elektronika Negeri Surabaya (PENS) Institut Teknologi Sepuluh Nopember (ITS).